

CLAIMS

WE CLAIM:

1. A safety industrial controller for executing a safety program, the controller executing a stored program to:
 - (i) download safety program data to a memory of the controller;
 - (ii) read the safety program data in memory to derive a signature functionally
 - 5 dependant on values of the safety program data in memory; and
 - (iii) compare the signature to a stored signature derived from previously certified safety program data.
2. The safety industrial controller of claim 1 wherein the controller further executes the stored program to upload a representation of the safety program data as stored in memory.
3. The safety industrial controller of claim 2 wherein the controller further executes the stored program to store a copy of the representation of the safety program data as stored in memory in a separate portion of memory.
4. The safety industrial controller of claim 1 wherein the controller further executes the stored program to:
 - block execution of the safety program in memory when the derived signature does not match the stored signature.
5. The safety industrial controller of claim 1 wherein the controller further executes the stored program to block the execution of the safety program in memory when the derived signature does not match the stored signature only when the safety program data is indicated to be certified program data and otherwise not blocking
- 5 the execution of the safety program in memory and indicating that the safety program in memory requires certification.

6. The safety industrial controller of claim 1 wherein the controller further executes the stored program to output an indication to a user when the derived signature does not match the stored signature.

7. The safety industrial controller of claim 1 wherein the controller further executes the stored program to output a copy of the signature to a user for recordation.

8. The safety industrial controller of claim 1 wherein the safety program data includes executable instructions and data providing arguments to the executable instructions.

9. The safety industrial controller of claim 1 wherein the stored signature is received with the downloaded safety program data.

10. The safety industrial controller of claim 1 wherein the signature is derived using a cyclic redundancy code taking the safety program data as an argument.

11. The safety industrial controller of claim 1 wherein the cyclic redundancy code is selected to provide less than 2×10^{-8} possibility of an undetected difference between the safety program data in memory and certified safety program data used to generate the stored signature.

12. The safety industrial controller of claim 1 wherein the controller further executes to receiving standard program data to the memory of the controller; and wherein the signature is functionally independent of the standard program data.

13. The safety industrial controller of claim 1 wherein the controller includes two processors having associated portions of memory and wherein the controller further executes the stored program to:

in step (i) to download safety program data to both portions of memory of
5 the two processors;

in step (ii) to read the safety program data in both portions of memory to derive a signature functionally dependant on values of the safety program data in both portions of memory; and

10 in step (iii) compare the signature to a stored signature derived from previously certified safety program data executing on the controller in both portions of memory.

14. A safety industrial controller for executing a safety program, the controller comprising:

a means for receiving the safety program data to a memory of the controller;

5 a signature generator reading the safety program data in memory to derive a signature functionally dependant on values of the safety program data in memory; and

means for comparing a signature generated by the signature generator to a stored signature derived from previously certified safety program data.

15. A method of operating a safety industrial controller comprising the steps of:

(a) creating a safety program;

(b) downloading the safety program data to memory of a safety controller;

5 (c) certifying operation of the downloaded safety program executing on the safety controller;

(d) creating a first signature of the downloaded and certified safety program, the signature functionally dependent on values of the safety program data in memory; and

10 (e) storing the first signature.

16. The method of claim 15 further including the steps of:

(f) re-downloading the safety program data

(g) creating a second signature of the re-downloaded safety program data, the signature functionally dependant on values of the safety program data in memory;

5 and

(h) comparing the first signature to the second signature to establish that the re-downloaded safety program does not need to be re-certified.

17. The method of claim 15 wherein step (d) includes the step of creating a representation of the safety program data as stored in memory and uploading the representation to a user;

and wherein step (f) downloads the representation.

18. The method of claim 17 wherein the representation is a memory image of the safety program data.

19. The method of claim 15 further including the step of executing the downloaded program only if the stored signature matches the signature of the reloaded program.

20. The method of claim 16 wherein:

step (b) downloads the safety program data to separate portions of memory associated with two independent processors;

5 step (c) certifies operation of the downloaded safety programs executing in parallel on the two processors;

step (d) creates the first signature of the downloaded and certified safety programs, the signature functionally dependent on values of the safety program data in both of the separate portions of memory;

10 step (f) re-downloads the safety program data to the separate portions of memory;

step (g) creating the second signature of the re-downloaded safety program data, the signature functionally dependant on values of the safety program data in both of the separate portions of memory.

21. The method of claim 15 further including the step of downloading standard program data and wherein steps (d) and (g) do not include the standard program data in the creation of the first and second signatures.